```
                                ;
                                ;       Version 1.1, Modified Dec 28, 1981 for 8086
                                ;
                                ;       Copyright (c) 1976, 1977, 1978, 1979, 1980, 1981
                                ;       Digital Research
                                ;       Box 579, Pacific Grove,
                                ;       California, 93950
                                ;
                                        cseg
                                        org     000h
                                ccploc:
0B00                            bdosl   equ     ccploc+0b00h
0800                            ccpcode equ     800h
0000                            false   EQU     0000h
FFFF                            true    EQU     not false
0000                            testing EQU     false                   ;true if debugging
0003                            ctlc    equu    03h                     ;control-c
0031                            asone   equ     031h                    ;ascii zero
                                ;
                                ;
                                ;       ************************************************************
                                ;       *       Base of CCP contains the following code        *
                                ;       *       ccp:    jmp ccpstart    (start with command)   *
                                ;       *               jmp ccpclear    (start, clear command) *
                                ;                       jmp ccpret              ;next - no system init
                                ;       *       ccp+6   127             (max command length)   *
                                ;       *       ccp+7   comlen          (command length = 00)  *
                                ;       *       ccp+8   ' ... '         (16 blanks)            *
                                ;       ************************************************************
                                ;       * Normal entry is at ccp, where the command line given *
                                ;       * at ccp+8 is executed automatically (normally a null  *
                                ;       * command with comlen = 00).  An initializing program  *
                                ;       * can be automatically loaded by storing the command   *
                                ;       * at ccp+8, with the command length at ccp+7.  In this *
                                ;       * case, the ccp executes the command before prompting  *
                                ;       * the console for input.  Note that the command is exe-*
                                ;       * cuted on both warm and cold starts.  When the command*
                                ;       * line is initialized, a jump to "jmp ccpclear" dis-   *
                                ;       * ables the automatic command execution.               *
                                ;       ************************************************************
                                ;
0000 E92A03         032D        JMP     ccpstart                ;start ccp with possible initial command
0003 E92103         0327        JMP     ccpclear                ;clear the command buffer
0006 E90203         030B        jmp     ccpret
0009 7F                 maxlen  db      127                     ;max buffer length
000A 00                 comlen  db      0                       ;command length (filled in by dos)
                                ;       (command executed initially if comlen non zero)
000B 202020202020       combuf  db      '       '               ;8 character fill
     2020
0013 202020202020               db      '       '               ;8 character fill
     2020
```

CP/M 86 V.1.1 - vol I
COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. # C81 - 162

```
001B 434F50595249          db      'COPYRIGHT (C) 1980, DIGITAL RESEARCH '       ; 38
     474854202843
     292031393830
     2C2044494749
     54414C205245
     534541524348
     2020
0041                       rs      74              ;8 + 8 + 74 = 128
                   ;       total buffer length is 128 characters
                   ;
                   ;
0001               rcharf  EQU     1                       ;read character function
0002               pcharf  EQU     2                       ;print character function
0009               pbuff   EQU     9                       ;print buffer function
000A               rbuff   EQU     10                      ;read buffer function
000B               breakf  EQU     11                      ;break key function
000C               liftf   EQU     12                      ;lift head function (no operation)
000D               initf   EQU     13                      ;initialize bdos function
000E               self    EQU     14                      ;select disk function
000F               openf   EQU     15                      ;open file function
0010               closef  EQU     16                      ;close file function
0011               searf   EQU     17                      ;search for file function
0012               searnf  EQU     18                      ;search for next file function
0013               delf    EQU     19                      ;delete file function
0014               dreadf  EQU     20                      ;disk read function
0015               dwritf  EQU     21                      ;disk write function
0016               makef   EQU     22                      ;file make function
0017               renf    EQU     23                      ;rename file function
0018               logf    EQU     24                      ;return login vector
0019               cself   EQU     25                      ;return currently selected drive number
001A               dmaf    EQU     26                      ;set dma address
0020               userf   EQU     32                      ;set user number
0032               biosf   equ     50              ;direct BIOS call
0033               dmabasf equ     51              ;set segment for dma disk
003A               initmaf equ     58      ;clear all memory allocations
                   ;
                   ;       special fcb flags
0009               rofile  EQU     9                       ;read only file
000A               sysfile EQU     10                      ;system file flag
                   ;
                   ;       special characters
0021               crr     equ     33              ;carriage return for NUMBER
000D               cr      EQU     13              ;carriage return
000A               lf      EQU     10              ;line feed
005F               la      EQU     5fh             ;left arrow
001A               eofile  EQU     1ah             ;end of file
                   ;
                   ;
                   ;
                   ;
                   ;       page0 in user data area
005C               userfcb equ     05ch
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
                        ;
                        ;       utility procedures
                        bdos:
008B CDE0                       int     224
008D 8CCD                       mov     bp,cs
008F 8EC5                       mov     es,bp
0091 C3                         ret
                        funct0:
0092 B100                       mov     cl,0            ;function zero in bdos for fret mem and restart
0094 33D2                       xor     dx,dx
0096 EBF3       008B            jmps    bdos
                        ;
                        printchar:
0098 8AD0                       MOV     DL,AL
009A B102                       MOV     CL,pcharf
009C EBED       008B            jmps    bdos
                        ;
                        blank:
009E B020                       MOV     AL,' '
                        ;
                        printbc:
                                                        ;print character, but save b,c registers
00A0 51                         PUSH    CX
00A1 E8F4FF     0098            CALL    printchar
00A4 59                         POP     CX
00A5 C3                         RET
                        ;
                        crlf:
00A6 B00D                       MOV     AL,cr
00A8 E8F5FF     00A0            CALL    printbc
00AB B00A                       MOV     AL,lf
00AD EBF1       00A0            jmps    printbc
                        ;
                        ;
                        print:                          ;print string starting at b,c until next 00 e
00AF 51                         PUSH    CX
00B0 E8F3FF     00A6            CALL    crlf
00B3 5B                         POP     BX              ;now print the string
00B4 8A07               prin0:  MOV     AL,[BX]
00B6 0AC0                       OR      AL,AL
00B8 7501       00BB            JNZ     L_1
00BA C3                         RET
                        L_1:                            ;stop on 00
00BB 43                         INC     BX
00BC 53                         PUSH    BX              ;ready for next
00BD E8D8FF     0098            CALL    printchar
00C0 5B                         POP     BX              ;character printed
00C1 EBF1       00B4            jmps    prin0           ;for another character
                        ;
                        ;
                        ;
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
                        lodfile:
00C3 BA2708                     mov     dx,offset comfcb
00C6 B13B                       mov     cl,59
00C8 EBC1       008B            jmps    bdos
                        ;
                        ;
                        ;
                        biosfunc:
00CA B93200                     mov     cx,biosf                ;direct BIOS
00CD BA3009                     mov     dx,offset biosparm
00D0 EBB9       008B            jmps    bdos
                        ;
                        initmem:
00D2 B13A                       mov     cl,initmaf              ;initialize bdos memory managemt
00D4 EB35       008B            jmps    bdos
                        initialize:
00D6 B10B                       MOV     CL,initf
00D8 EBB1       008B            jmps    bdos
                        ;
                        select:
00DA 8AD0                       MOV     DL,AL
00DC B10E                       MOV     CL,self
00DE EBAB       003B            jmps    bdos
                        ;
                        bosinr:
00E0 EBA8FF     008B            CALL    bdos
00E3 A2CD03                     MOV     dcnt,AL
00E6 FEC0                       INC     AL
00E8 C3                         RET
                        ;
                        open:                   ;open the file given by d,e
00E9 B10F                       MOV     CL,openf
00EB EBF3       00E0            jmps    bosinr
                        ;
                        openc:                                  ;open comfcb
00ED C606470800                 MOV     comrec,0                ;clear next record to read
00F2 BA2708                     mov     dx,offset comfcb
00F5 EBF2       00E9            jmps    open
                        ;
                        close:                  ;close the file given by d,e
00F7 B110                       MOV     CL,closef
00F9 E8E4FF     00E0            call    bosinr
00FC 7401       00FF            jz      closeerr
00FE C3                         ret
                        closeerr:
00FF B99B09                     mov     cx,offset closem
0102 EB20       0124            jmps    errm
                        ;
                        ;
                        srch:                   ;search for the file given by d,e
0104 B111                       MOV     CL,searf
0106 EBD8       00E0            jmps    bosinr
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
                        ;
                        srchn:
                                                ;search for the next occurrence of the file given by d,e
0108 B112                       MOV     CL,searnf
010A EBD4       00E0            jmps    bosinr
                        ;
                        srchcm:
                                                                        ;searct for comfcb file
010C BA2708                     mov     dx,offset comfcb
010F EBF3       0104            jmps    srch
                        ;
                        delete:                                         ;delete the file given by d,e
0111 B113                       MOV     CL,delf
0113 EB1D       0132            jmps    to_bdos2
                        ;
                        oscond:
0115 E873FF     008B            CALL    bdos
0118 0AC0                       OR      AL,AL
011A C3                         RET
                        ;
                        diskr:
                                                                        ;read the next record from the file given by
011B B114                       MOV     CL,dreadf
011D EBF6       0115            jmps    oscond
                        ;
                        diskrc:
                                                                        ;read the comfcb file
011F BA2708                     mov     dx,offset comfcb
0122 EBF7       011B            jmps    diskr
                        ;
                        ;
                        errm:
0124 E888FF     00AF            call    print
0127 E7B706     07E1            jmp     retcom
                        ;
                        ;
                        ;
                        renam:                                          ;rename the file given by d,e
012A B117                       MOV     CL,renf
012C EB04       0132            jmps    to_bdos2
                        ;
                        getuser:
                                                                        ;return current user code in a
012E B2FF                       MOV     DL,0ffh                         ;drop through to setuser
                        ;
                        setuser:
0130 B120                       MOV     CL,userf
                        to_bdos2:
0132 E956FF     008B            JMP     bdos    ;sets user number
                        ;
                        ;
                        translate:
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER #_____

```
                                                        ;translate character in register A to upper c
0135 3C61                       CMP     AL,61h
0137 7206       013F            jb      lret4
                                        ;return if below lower case a
0139 3C7B                       CMP     AL,7bh
013B 7302       013F            jnb     lret4
013D 245F                       AND     AL,5fh
013F C3                 lret4:  ret                     ;translated to upper case
                        ;
                        readc:
                        ;
0140 F6060508FF                 test    submit,0ffh             ;check for submit
0145 7458       019F            jz      nosub
                                                        ;scanning a submit file
0147 F6063909FF                 test    cdisk,0ffh              ;check for drive A
014C 7405       0153            jz      readca
014E 33C0                       xor     ax,ax
0150 E887FF     00DA            call    select
                        readca:
0153 BA0608                     mov     dx,offset subfcb
0156 E890FF     00E9            call    open                    ;open again for xsub
0159 7444       019F            jz      nosub           ;skip if no sub
015B A01508                     mov     al,subrc
015E FEC8                       dec     al
0160 A22608                     mov     subcr,al
0163 BA0608                     mov     dx,offset subfcb
0166 E8B2FF     011B            call    diskr                   ;end of file if last record
0169 7534       019F            jnz     nosub
                        ;
                                        ;disk read is ok, transfer to combuf
016B BF0A00                     mov     di,offset comlen
016E BE4B08                     mov     si,offset ccpbuff
0171 B98000                     mov     cx,128
0174 E8C502     043C            call    move0                   ;line transferred
                                                        ;deleted record
0177 C606140800                 mov     submod,0h               ;clear fwflag....S2 byte
017C FE0E1508                   dec     subrc                   ;one less record
0180 BA0608                     mov     dx,offset subfcb
0183 E871FF     00F7            call    close                   ;close file with deleted record
0186 7417       019F            jz      nosub
                                                        ;close went ok, return to original drive
0188 A03909                     mov     al,cdisk                ;now select current disk
018B E84CFF     00DA            call    select
018E BB0B00                     mov     bx,offset combuf
0191 E820FF     00B4            call    prin0                   ;print command buffer to 00h
0194 E83200     01C9            call    breakk
0197 7411       01AA            jz      noread
0199 E85600     01F2            call    delsub
019C E9BF01     035E            jmp     ccp                     ;break key depressed
                                                        ;
                        nosub:                          ;no submit file
019F E85000     01F2            call    delsub
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950

```
                                        ;read the next command into the command buffer
                                        ;translate to upper case, store zero at end
01A2 B10A                       mov     cl,rbuff
01A4 BA0900                     mov     dx,offset maxlen
01A7 E8E1FE     008B            call    bdos
                noread:
01AA BB0B00                     mov     bx,offset comlen+1      ;command buffer in BX?
01AD 33C9                       xor     cx,cx
01AF 2E8A0E0A00                 mov     cl,comlen
01B4 E30A       01C0            jcxz    readc1
                readc0:
01B6 8A07                       mov     al,byte ptr[bx]         ;get char and translate
01B8 E87AFF     0135            call    translate
01BB 8807                       mov     byte ptr[bx],al         ;store in buffer
01BD 43                         inc     bx
01BE E2F6       01B6            loop    readc0
                                                                ;
                readc1:                                         ;end of scan, bx address end of command
01C0 882F                       mov     byte ptr[bx],ch         ;zero to end of command
01C2 C70635090B00               mov     comaddr,offset combuf
01C8 C3                         ret
                ;
                breakk:
                                                                ;check for a character ready at the console
01C9 B10B                       MOV     CL,breakf
01CB E8BDFE     008B            CALL    bdos
01CE 0AC0                       OR      AL,AL
01D0 7501       01D3            JNZ     L_6
01D2 C3                         RET
                L_6:
01D3 B101                       MOV     CL,rcharf
01D5 E8B3FE     008B            call    bdos    ;character cleared
01D8 0AC0                       OR      AL,AL
01DA C3                         RET
                ;
                cselect:
                                                                ;get the currently selected drive number to r
01DB B119                       MOV     CL,cself
01DD EB10       01EF            JMps    jp_bdos3
                setdmab:
                                        ;set dma segment base
01DF B133                       mov     cl,dmabasf
01E1 EB0C       01EF            jmps    jp_bdos3
                ;
                setbufu:
                                        ;set dma address to default buffer in user
                                        ;data group
01E3 8B160308                   mov     dx,pag0
01E7 E8F5FF     01DF            call    setdmab
01EA BA8000                     MOV     DX,offset userbuff              ;(drop through)
                ;
                setdma:
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
                                                        ;set dma address to d,e
01ED B11A                       MOV     CL,dmaf
                        jp_bdos3:
01EF E999FE     003B            JMP     bdos
                        ;
                        ;
                        ;
                        ;
                        ;
                        delsub:
                                                        ;delete the submit file, and set submit flag
01F2 F6060508FF                 test    submit,0ffh
01F7 7501       01FA            jnz     delsub1
01F9 C3                         ret                     ;return if no sub file
                        delsub1:
01FA B000                       mov     al,0h
01FC A20508                     mov     submit,al               ;submit flag is set to false
01FF E8D8FE     00DA            call    select          ;on drive a to erase file
0202 BA0608                     mov     dx,offset subfcb
0205 E809FF     0111            call    delete
0208 A03909                     MOV     AL,cdisk
020B E9CCFE     00DA            JMP     select          ;back to original drive
                        ;
                        serize:
                                                        ;check serialization
020E BFDC09                     mov     di,offset serial
0211 BE000B                     MOV     si,OFFSET(bdos1)
0214 B90600                     MOV     cx,6            ;check six bytes
0217 F3A6                repz   cmps    al,al
0219 E302       021D            jcxz    ser0
021B EB01       021E            jmps    badserial
021D C3                 ser0:   RET                     ;serial number is ok
                        ;
                        badserial:
021E F4                         HLT                     ;for now
                                                        ;
                        ;
                        ;
                        ; fcb scan and fill subroutine (entry is at fillfcb below)
                                                        ;fill the comfcb, indexed by A (0 or 16)
                                                        ;subroutines
                        delim:
021F 8A04                       mov     al,[si]
0221 0AC0                       or      al,al
0223 7501       0226            jnz     delim0
0225 C3                         ret                     ;return with Z flag set if string end
                        delim0:
0226 3C20                       cmp     al,' '          ;check for legal character
0228 7303       022D            jae     delim1
022A E99301     03C0            jmp     comer           ;not graphic
                        delim1:
022D 57                         push    di
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
                                ;
022E BF3B09                     mov     di,offset delims
0231 B90900                     mov     cx,9            ;number of delimiters
0234 FC                         cld
0235 F2AE               repne   scas    al
0237 5F                         pop     di
0238 C3                         ret                     ;return with Z flag set if delimiter
                        ;
  0020                  blanke  equ     20h
                        ;
                        deblank:
                        ;       move SI pointer to first non-blank character in stream
                        ;       non-blank character returned in AL
                        ;
0239 8A04                       mov     al,[si]
023B 3C00                       cmp     al,00h          ;end of string
023D 7501               0240    jnz     deblank0
023F C3                         ret                     ;Z flag set if end of string
                        deblank0:
0240 3C20                       cmp     al,blanke
0242 7401               0245    jz      deblank1
0244 C3                         ret                     ;Z flag off if non-blank found
                        deblank1:
0245 46                         inc     si
0246 EBF1               0239    jmps    deblank
                        ;
                        ;

                        padfield:
                        ;       pad field with number of blanks specified in CX
                        ;
0248 43                         inc     bx
0249 C60720                     mov     byte ptr[bx],' '
024C E2FA               0248    loop    padfield
024E C3                         ret
                        ;
                        getident:
                        ;       move identifier from [SI] to [BX]...truncate all characters
                        ;       after number indicated in CX.  if identifier contains fewer
                        ;       characters, pad field in [BX] with blanks
                        ;
024F 51                         push    cx
0250 E8CCFF             021F    call    delim
0253 59                         pop     cx
0254 74F2               0248    jz      padfield
0256 43                         inc     bx
0257 3C2A                       cmp     al,'*'          ;pad remainder of field with ?
0259 7504               025F    jnz     getid1
025B B03F                       mov     al,'?'          ;char to store is ?
025D EB01               0260    jmps    getid2
                        getid1:
025F 46                         inc     si
                        getid2:
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER #

```
                                        ;store char in [BX]
0260 8807                       mov     [bx],al
0262 E2EB       024F            loop    getident
                trid:                   ;truncate identifier > required length
0264 E8B3FF     021F            call    delim
0267 7403       026C            jz      endgetid
0269 46                         inc     si
026A EBF8       0264            jmps    trid
                endgetid:
026C C3                         ret
                ;
                ;
                ;
                ;  parse file control block
                ;
                ; on input SI contains address of command string.....comaddr
                ; on input BX contains address of file control block
                ;
                filfcb0:
026D B80000                     mov     ax,0h           ;equivalent to fillfcb(0)
                ;
                filfcb:
0270 BB2708                     mov     bx,offset comfcb        ;offset of command FCB to fill
0273 03D8                       add     bx,ax           ;AX contains 16 for second FCB
0275 53                         push    bx              ;fcb rescanned at end of procedure
0276 C6063A0900                 mov     sdisk,0h        ;clear selected disk
027B 8B363509                   mov     si,comaddr      ;address of string to parse
027F E8B7FF     0239            call    deblank         ;move to first non-blank character
0282 89363709                   mov     staddr,si       ;save address for error messages
0286 7408       0290            jz      setcr0          ;deblank sets Z flag if end string(0h)
0288 2C40                       sbb     al,'A'-1        ;change to binary
028A 807C013A                   cmp     byte ptr 1[si],':'      ;check for disk ID
028E 7405       0295            jz      setdsc
                setcr0:
0290 A03909                     mov     al,cdisk        ;use current disk
0293 EB12       02A7            jmps    setdscx
                setdsc:
0295 3C10                       cmp     al,16
0297 7704       029D            ja      filer
0299 3C00                       cmp     al,0
029B 7705       02A2            ja      setdsc1         ;disk must be A through P
                ;
                filer:
029D B4FF                       mov     ah,0ffh
029F 5B                         pop     bx
02A0 EB39       02DB            jmps    filfexit
                ;
                setdsc1:
02A2 A23A09                     mov     sdisk,al        ;mark as selected
02A5 46                         inc     si
02A6 46                         inc     si              ;mov to first char of name
                setdscx:
02A7 8807                       mov     [bx],al         ;disk number to FCB
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
                        ;
                        ;  get filename
                        ;
02A9 B90800                     mov     cx,8            ;number of character in parse field
02AC E8A0FF     024F            call    getident        ;return with name padded with blanks
                        ;
                        ;  get type
                        ;
02AF B90300                     mov     cx,3
02B2 3C2E                       cmp     al,'.'          ;was delimiter a '.'
02B4 7501       02B7            jnz     setty1
02B6 46                         inc     si              ;past the '.'
                        setty1:
02B7 E895FF     024F            call    getident        ;return with 3 position type field
                        ;
                        ;
                        ;  fill extent and system byte with zero
                        ;
                        efill:
02BA 43                         inc     bx
02BB 8BFB                       mov     di,bx
02BD 8BC1                       mov     ax,cx           ;zero to AL
02BF B90300                     mov     cx,3
02C2 FC                         cld
02C3 33C0                       xor     ax,ax
02C5 F3AA                rep    stos    al
02C7 893635C9                   mov     comaddr,si              ;set new starting point
                        ;
                        ;    recover the start address of the fcb and count ?'s
                        ;
02CB 5B                         pop     bx
02CC B90800                     mov     cx,11
                        scnq:
02CF 43                         inc     bx
02D0 803F3F                     cmp     byte ptr[bx],'?'
02D3 7502       02D7            jnz     scnq0
02D5 FEC4                       inc     ah              ;? found, count it in AH
02D7 E2F6       02CF    scnq0:  loop    scnq
                        ;
02D9 8A04                       mov     al,[si]         ;last character processed
                        filfexit:
02DB 0AE4                       or      ah,ah           ;set zero flag
02DD C3                         ret
                        ;
                        ;
                        intrinsic:
                                                        ;look for intrinsic functions (comfcb has bee
                        ;
02DE BF4409                     mov     di,offset intvec
02E1 B80600                     mov     bx,intlen
                        intr1:
02E4 8E2808                     mov     si,offset comfcb+1
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
                                
02E7 B90400                     mov     cx,4
02EA F3A6                 rep   cmps    al,al
02EC 7408           02F6        jz      intr2
02EE 03F9                       add     di,cx
02F0 4B                         dec     bx
02F1 75F1           02E4        jnz     intr1
                          intr11:
02F3 B007                       mov     al,7
02F5 C3                         ret
                          intr2:
02F6 8A04                       mov     al,[si]         ;check command is followed by blank
02F8 3C20                       cmp     al,blanke
02FA 75F7           02F3        jnz     intr11
02FC 8BC3                       mov     ax,bx
02FE C3                         ret
                        ;
                        ;
                        ;
                          set_ccp_dma:
02FF 8CCA                       mov     dx,cs           ;set dma base
0301 E8DBFE         01DF        call    setdmab
0304 BA4B08                     mov     dx,offset ccpbuff
0307 E8E3FE         01ED        call    setdma
030A C3                         ret
                        ;
                        ;
                        ;
                        ;
                        ;       return from user function
                          ccpret:
030B C70635090B00               mov     comaddr,offset combuf           ;reset ptr to combuf
0311 2E803E0B0003               cmp     byte ptr combuf,ctlc
0317 7506           031F        jnz     ccpret1
0319 2EC6060A0000               mov     comlen,0h               ;BDOS does not reset len for ctl-c
                          ccpret1:
031F BC3009                     mov     sp,offset stack_base
0322 E83E01         0463        call    resetdu         ;reset disk and user
0325 EB2C           0353        jmps    ccpstart1       ;check for chaining
                        ;
                          ccpclear:
                                                                ;clear the command buffer
0327 2EC6060A0000               MOV     byte ptr comlen,0h
                                                        ;drop through to start ccp
                          ccpstart:
                                                        ;enter here from boot loader
032D BC3009                     mov     sp,offset stack_base
0330 51                         PUSH    CX                      ;save initial disk number
                                                        ;(high order 4bits=user code, low 4bits=disk#
0331 8AC1                       MOV     AL,CL
0333 B104                       mov     cl,4
0335 D2D8                       rcr     al,cl
0337 240F                       AND     AL,0fh                  ;user code
```

```
                                MOV     DL,AL
0339 8AD0                       mov     cuser,al            ;save user number ..restore
033B A2CE08                                                 ;after program abort
033E E8EFFD     0130            CALL    setuser             ;user code selected
                                                            ;initialize for this user, get $ flag
0341 E892FD     00D6            CALL    initialize          ;0ffh in accum if $ file present
0344 A20508                     mov     submit,al           ;submit flag set of $ file present
0347 E883FD     00D2            call    initmem             ;initialize bdos memory management
034A 58                         pop     ax                  ;recall user code and disk number
034B 240F                       and     al,0fh              ;disk number only
034D A23909                     mov     cdisk,al            ;clears user code nibble
0350 E837FD     00DA            call    select              ;proper disk is selected, now check sub files
                            ;
                            ccpstart1:
                                                            ;get os base for dma address
0353 E8A9FF     02FF            call    set_ccp_dma
                                                            ;check for initial command
0356 2E803E0A0000               cmp     comlen,0h
035C 751A       0378            jnz     ccp0                ;assume typed already
                            ;
                            ccp:
                                                            ;enter here on each command or error conditio
035E FC                         cld
035F BC3009                     mov     sp,offset stack_base
0362 E841FD     00A6            CALL    crlf                ;print d> prompt, where d is disk name
0365 E873FE     01DB            CALL    cselect             ;get current disk number
0368 0441                       ADD     AL,'A'
036A E82BFD     0098            CALL    printchar
036D B03E                       MOV     AL,'>'
036F E826FD     0098            CALL    printchar
0372 E88AFF     02FF            call    set_ccp_dma
0375 E8C9FD     0140            CALL    readc               ;command buffer filled
                            ccp0:                           ;(enter here from initialization with command
0378 FC                         cld
0379 E85FFE     01DB            CALL    cselect
037C A23909                     MOV     cdisk,AL            ;current disk number saved
037F 2E803E0300038              cmp     byte ptr combuf,';' ;this could be comment
0385 74D7       035E            jz      ccp
0387 8B363509                   mov     si,comaddr          ;check blank lines
038B E8ABFE     0239            call    deblank
038E 74CE       035E            jz      ccp                 ;blank line
0390 E8DAFE     026D            CALL    filfcb0             ;command fcb filled
0393 752B       03C0            jnz     comer
                                                            ;the name cannot be an ambiguous reference
0395 F6063A09FF                 test    sdisk,0ffh          ;was disk id specified in command
039A 7403       039F            jz      ccp2
039C E94503     06E4            jmp     userfunc
                            ccp2:
039F 803E280820                 cmp     byte ptr comfcbt1,' '   ;was a command specified
03A4 741A       03C0            jz      comer
                                                            ;check for an intrinsic function
03A6 E835FF     02DE            CALL    intrinsic
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER. # _____

```
690
691  03A9 48                    dec     ax
692  03AA 8BE209                mov     bx,offset jmptab        ;index is in the accumulator
693  03AD 03D8                  add     bx,ax
694  03AF 03D8                  add     bx,ax
695                                                             ;index in d,e
696  03B1 8B17                  mov     dx,[bx]
697  03B3 FFE2                  JMP     dx
698                         ;pc changes to the proper intrinsic or user function
699                                                             ;utility subroutines for intrinsic handlers
700                         readerr:
701                                                             ;print the read error message
702  03B5 B99009                mov     cx,offset rdmsg
703  03B8 EB03          03BD    JMps    to_print
704
705                         nofile:
706                                                             ;print no file message
707  03BA B95C09                mov     cx,offset nofmsg
708                         to_print:
709  03BD E9EFFC        00AF    JMP     print
710                         comer:
711                                                 ;error in command string starting at position
712                                                             ;'staddr' and ending with first delimiter
713  03C0 E8E3FC        00A6    call    crlf                    ;space to next line
714  03C3 8B1E3709              mov     bx,word ptr staddr      ;bx addresses first to print
715                         comer0:                             ;print characters until blank or zero
716  03C7 8A07                  mov     al,byte ptr[bx]
717  03C9 3C20                  cmp     al,' '
718  03CB 740C          03D9    jz      comer1          ; not blank
719  03CD 0AC0                  or      al,al
720  03CF 7408          03D9    jz      comer1          ; not zero, so print it
721  03D1 53                    push    bx
722  03D2 E8C3FC        0098    call    printchar
723  03D5 5B                    pop     bx
724  03D6 43                    inc     bx
725  03D7 EBEE          03C7    jmps    comer0                  ; for another character
726                         comer1:                             ;print question mark,and delete sub file
727  03D9 B03F                  mov     al,'?'
728  03DB E8BAFC        0098    call    printchar
729  03DE E8C5FC        00A6    call    crlf
730  03E1 2EC6060A0000          mov     comlen,0h               ;zero command len
731  03E7 E974FF        035E    jmp     ccp
732                         ;
733                         getnumber:                          ;read a number from the command line
734  03EA E880FE        026D    CALL    filfcb0                 ;should be number
735  03ED F6063A09FF            test    byte ptr sdisk,0ffh
736  03F2 75CC          03C0    jnz     comer
737                                                             ;cannot be prefixed
738                                                             ;convert the byte value in comfcb to binary
739  03F4 BB2808                mov     bx,offset comfcb+1
740  03F7 B90B00                MOV     CX,11                   ;(b=0, c=11)
741  03FA 803F20                cmp     byte ptr[bx],' '                ;check for blanks only
742  03FD 7503          0402    jnz     conv0
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
                                
03FF B021                       mov     al,crr          ;return carriage return in AL
0401 C3                         ret
                                                        ;value accumulated in b, c counts name length
0402 8A07               conv0:  MOV     AL,BYTE PTR [BX]
0404 3C20                       CMP     AL,' '
0406 7427               042F    JZ      conv1
                                                        ;more to scan, convert char to binary and add
0408 43                         INC     BX
0409 2C30                       SUB     AL,'0'
040B 3C0A                       CMP     AL,10
040D 73B1               03C0    jae     comer           ;valid?
040F 8AF0                       MOV     DH,AL           ;save value
0411 F6C5E0                     test    CH,11100000b
0414  5AA               03C0    jnz     comer
0416 8AC5                       MOV     AL,CH           ;recover value
0418 B103                       mov     cl,3
041A D2C0                       rol     al,cl
                                                                ;*8
041C 02C5                       ADD     AL,CH
041E 72A0               03C0    jb      comer
0420 02C5                       ADD     AL,CH
0422 729C               03C0    jb      comer
                                                                ;*8+*2 = *10
0424 02C6                       ADD     AL,DH
0426 7298               03C0    jb      comer
                        L_29:                           ;+digit
0428 8AE8                       MOV     CH,AL
042A FEC9                       DEC     CL
042C 75D4               0402    JNZ     conv0           ;for another digit
042E C3                         RET
                        conv1:                          ;end of digits, check for all blanks
042F 803F20                     cmp     byte ptr [bx],' '
0432 758C               03C0    jnz     comer
                                                        ;blanks?
0434 43                         INC     BX
0435 FEC9                       DEC     CL
0437 75F6               042F    JNZ     conv1
0439 8AC5                       MOV     AL,CH           ;recover value
043B C3                         RET
                                                        ;
                        move0:
043C FC                         cld
043D F3A4                rep    movs    al,al
043F C3                         ret
                                                        ;
                        addhcf:                         ;buff + a + c to bx followed by fetch
0440 BB4B08                     mov     bx,offset ccpbuff
0443 B400                       mov     ah,0
0445 02C1                       add     al,cl
0447 03D8                       add     bx,ax
0449 8A07                       MOV     AL,BYTE PTR [BX]
```

cp/m 86 V.1.1-vol I
COPYRIGHT © 1981, 198
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 939
SER # C81-162

```
                                                                                
                                                                                
044B C3                         RET                                             
                                                                                
                                                        ;                       
                        setdsk:                                                 
                                                        ;change disks for this command, if requested
044C C606270800                 mov     byte ptr comfcb,0h              ;clear disk name from fcb
0451 A03A09                     MOV     AL,sdisk                                
0454 0AC0                       OR      AL,AL                                   
0456 7424               047C    jz      l33ret                                  
                                                        ;no action if not specified
0458 FEC8                       DEC     AL                                      
045A BB3909                     mov     bx,offset cdisk                         
045D 3A07                       CMP     AL,BYTE PTR [BX]                        
045F 741B               047C    jz      l33ret                                  
                                                        ;already selected       
0461 EB1D               0480    JMps    jp_select                               
                        resetdu:                                                
                                ;                       reset disk number and user number after
                                ;                       user function           
0463 8A16CE08                   mov     dl,cuser                                
0467 E8C6FC             0130    call    setuser                                 
046A EB11               047D    jmps    reset1                                  
                                ;                                               
                                                        ;                       
                        resetdisk:                                              
                                                        ;return to original disk after command
046C A03A09                     mov     al,sdisk                                
046F 0AC0                       or      al,al                                   
0471 7409               047C    jz      l33ret          ;no action if not selected
0473 FEC8                       dec     al                                      
0475 BB3909                     mov     bx,offset cdisk                         
0478 3A07                       cmp     al, [bx]                                
047A 7501               047D    jnz     reset1                                  
047C C3                 l33ret: ret                     ;same disk              
                                ;                                               
047D A03909             reset1: mov     al,cdisk                                
                        jp_select:                                              
0480 E957FC             00DA    jmp     select                                  
                                ;                                               
                                                        ;individual intrinsics follow
                                ;                                               
                                ;                                               
                        directs:                                                
                                                        ;directory search       
0483 C606F10901                 mov     typedir,1h                              
0488 EB2700             04B2    call    dir                                     
048B 803EF00900                 cmp     anyfils,0h              ;any system files???
0490 7406               0498    jz      directs2                                
0492 B9AA09                     mov     cx,offset dirfiles      ;tell the user  
0495 EB17FC             00AF    call    print                                   
                        directs2:                                               
0498 EB15               04AF    jmps    direct2                                 
                                ;                                               
```

COPYRIGHT © 1981 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER #

```
                        direct:                                 ;directory search
049A C606F10900                 mov     typedir,0h
049F E81000     04B2            call    dir
04A2 803EF00900                 cmp     anyfils,0h              ;any system files???
04A7 7406       04AF            jz      direct2
04A9 B9C509                     mov     cx,offset sysfiles      ;tell the user
04AC E800FC     00AF            call    print
                        direct2:
04AF E972F03    07E1            jmp     retcom
                        ;
                        ;
                        ;
                        dir:
04B2 2EC6060A0000               mov     comlen,0h               ;user may abort with ctl-c
04B8 C606F00900                 mov     anyfils,0h              ;any non-selected files??
04BD E8ADFD     026D            CALL    filfcb0                 ;comfcb gets file name
04C0 80FCFF                     cmp     ah,0ffh                 ;invalid drive??
04C3 7503       04C8            jnz     dir0a
04C5 E9F8FE     03C0            jmp     comer
                        dir0a:
04C8 E881FF     044C            CALL    setdsk                  ;change disk drives if requested
04CB BB2808                     mov     bx,offset comfcb+1
04CE 803F20                     cmp     byte ptr [bx],' '
04D1 750A       04DD            JNZ     dir1            ;skip fill of ??? if not blank
                                                        ;set comfcb to all ??? for current disk
04D3 B50B                       MOV     CH,11           ;length of fill ????????.???
04D5 C6073F             dir0:   MOV     BYTE PTR [BX],'?'
04D8 43                         INC     BX
04D9 FECD                       DEC     CH
04DB 75F8       04D5            JNZ     dir0
                                                        ;not a blank request, must be in comfcb
04DD B200               dir1:   MOV     DL,0
04DF 52                         PUSH    DX              ;E counts directory entries
04E0 E829FC     010C            CALL    srchcm          ;first one has been found
04E3 7503       04E8            JNZ     dir2
04E5 E8D2FE     03BA            CALL    nofile
                                                ;not found message
04E8 7503       04ED    dir2:   jnz     dir21
04EA E99B00     0588            jmp     endir
                        dir21:
                                                        ;found, but may be system file
04ED A0CD03                     MOV     AL,dcnt         ;get the location of the element
04F0 B103                       mov     cl,3
04F2 D2C8                       ror     al,cl
                                                        ;c contains base index into buff for dir entr
04F4 8AC8                       mov     cl,al
04F6 B00A                       mov     al,sysfile
04F8 E845FF     0440            call    addhcf          ;value to A
04FB D0D0                       rcl     al,1
04FD 730E       050D            jnc     non_sys_files
                        ;       directory entry is for system file
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
902
903                             ;
904  04FF F606F109FF                    test    typedir,true        ;true if DIRS command
905  0504 7515          051B            jnz     dir25               ;yes for command
906  0506 C606F00901                    mov     anyfils,01h         ;sys file exists..non_system search
907  050B EB70          057D            jmps    dir6
908                             ;
909                             non_sys_files:
910  050D F606F109FF                    test typedir,true           ;not true if DIR
911  0512 7407          051B            jz      dir25               ;ok to list
912  0514 C606F00901                    mov     anyfils,01          ;tell DIRS non_sys files here
913  0519 EB62          057D            jmps    dir6
914                             ;
915                             ;
916                             dir25:                              ;skip if incorrect DIR attribute
917                                                                 ;c holds index into buffer
918                                                                 ;another fcb found, new line?
919  051B 5A                            POP     DX
920  051C 8AC2                          MOV     AL,DL
921  051E FEC2                          INC     DL
922  0520 52                            PUSH    DX
923                                                                 ;e=0,1,2,3,...new line if mod 4 = 0
924  0521 90                            nop
925  0522 90                            nop
926  0523 90                            nop
927  0524 2403                          AND     AL,11b
928  0526 50                            PUSH    AX                  ;and save the test
929  0527 7514          053D            JNZ     dirhd0          ;header on current line
930  0529 E87AFB        00A6            CALL    crlf
931  052C 51                            PUSH    CX
932  052D E8ABFC        01DB            CALL    cselect
933  0530 59                            POP     CX
934                                                                 ;current disk in A
935  0531 0441                          ADD     AL,'A'
936  0533 E86AFB        00A0            CALL    printbc
937  0536 B03A                          MOV     AL,':'
938  0538 E865FB        00A0            CALL    printbc
939  053B EB03          0545            jmps    dirhd1          ;skip current line hdr
940  053D E85EFB        009E dirhd0:    CALL    blank               ;after last one
941  0540 B03A                          MOV     AL,':'
942  0542 E85BFB        00A0            CALL    printbc
943                             dirhd1:
944  0545 E856FB        009E            CALL    blank
945                                                                 ;compute position of name in buffer
946  0548 B501                          MOV     CH,1                ;start with first character of name
947  054A 8AC5          dir3:   MOV     AL,CH
948  054C E8F1FE        0440            CALL    addhcf              ;buffta+c fetched
949  054F 247F                          AND     AL,7fh              ;mask flags
950                                                                 ;may delete trailing blanks
951  0551 3C20                          CMP     AL,' '
952  0553 7513          0568            JNZ     dir4            ;check for blank type
953  0555 58                            POP     AX
954  0556 50                            PUSH    AX                  ;may be 3rd item
```

SER. #

```
 955
 956  0557 3C03                  CMP     AL,3
 957  0559 750B          0566    JNZ     dirb            ;place blank at end if not
 958  055B B009                  MOV     AL,9
 959  055D E8E0FE        0440    CALL    addhcf                  ;first char of type
 960  0560 247F                  AND     AL,7fh
 961  0562 3C20                  CMP     AL,' '
 962  0564 7416          057C    JZ      dir5
 963                                                             ;not a blank in the file type field
 964  0566 B020          dirb:   MOV     AL,' '                  ;restore trailing filename chr
 965                     dir4:
 966  0568 E835FB        00A0    CALL    printbc                 ;char printed
 967  056B FEC5                  INC     CH
 968  056D 80FD0C                cmp     ch,12
 969  0570 730A          057C    JAE     dir5
 970                                                             ;check for break between names
 971  0572 80FD09                cmp     ch,9
 972  0575 75D3          054A    JNZ     dir3                    ;for another char
 973                                                             ;print a blank between names
 974  0577 E824FB        009E    CALL    blank
 975  057A EBCE          054A    jmps    dir3
 976                                                             ;
 977                     dir5:                                   ;end of current entry
 978  057C 58                    POP     AX                      ;discard the directory counter (mod 4)
 979  057D E849FC        01C9 dir6:  CALL    breakk              ;check for interrupt at keyboard
 980  0580 7506          0588    JNZ     endir           ;abort directory search
 981  0582 E883FB        0108    CALL    srchn
 982  0585 E960FF        04E8    jmp     dir2                    ;for another entry
 983                     endir:                                  ;end of directory scan
 984  0588 5A                    pop     dx                      ;discard directory counter
 985  0589 C3                    ret
 986                     ;
 987                     ;
 988                     erase:
 989  058A 2EC6060A0000          mov     comlen,0h               ;user may abort with ctl-c
 990  0590 E8DAFC        026D    CALL    filfcb0                 ;cannot be all ???'s
 991  0593 80FC0B                CMP     AH,11
 992  0596 7521          05B9    JNZ     erasefile
 993                                                             ;erasing all of the disk
 994  0598 B96409                mov     cx,offset ermsg
 995  059B E811FB        00AF    CALL    print
 996  059E E89FFB        0140    CALL    readc
 997  05A1 BB0A00                mov     bx,offset comlen
 998  05A4 FE0F                  DEC     BYTE PTR [BX]
 999  05A6 7403          05AB    JZ      L_38
1000  05A8 E983FD        035E    JMP     ccp
1001                     L_38:                                   ;bad input
1002  05AB 43                    INC     BX
1003  05AC 803F59                cmp     byte ptr [bx],'Y'
1004  05AF 7403          05B4    JZ      L_39
1005  05B1 E9AAFD        035E    JMP     ccp
1006                     L_39:
1007                                                             ;ok, erase the entire diskette
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
S/N: _____

```
1008
1009  05B4 43                    INC     BX
1010  05B5 891E3509              MOV     comaddr,BX          ;otherwise error at retcom
1011                       erasefile:
1012  05B9 803E280820            cmp     comfcb+1,' '        ;check for blank filename
1013  05BE 7467          0627    jz      jcomer
1014  05C0 E889FE        044C    CALL    setdsk
1015  05C3 BA2708                mov     dx,offset comfcb
1016  05C6 E848FB        0111    call    delete
1017  05C9 FEC0                  INC     AL                  ;255 returned if not found
1018  05CB 7503          05D0    jnz     eraret
1019                       call_nofile:
1020  05CD E8EAFD        03BA    call    nofile              ;no file message
1021                       eraret:
1022  05D0 EB4C          061E    jmps    jretcom
1023                       ;
1024                       ;
1025                       typec:
1026  05D2 2EC6060A0000          mov     comlen,0h           ;zero for re-enter
1027  05D8 E892FC        026D    CALL    filfcb0
1028  05DB 754A          0627    jnz     jcomer
1029                                                         ;don't allow ?'s in file name
1030  05DD 803E280820            cmp     comfcb+1,' '
1031  05E2 7443          0627    jz      jcomer
1032  05E4 E865FE        044C    CALL    setdsk
1033  05E7 E803FB        00ED    CALL    openc               ;open the file
1034  05EA 74E1          05CD    JZ      call_nofile         ;zero flag indicates not found
1035                                                         ;file opened, read 'til eof
1036  05EC E8B7FA        00A6    CALL    crlf
1037  05EF B8FF00                mov     bx,255              ;read first buffer
1038                       type0:                           ;loop on bptr
1039  05F2 81FB8000              cmp     bx,128
1040                                                         ;end buffer
1041  05F6 720A          0602    JB      type1
1042  05F8 53                    PUSH    BX                  ;carry if 0,1,...,127
1043                                                         ;read another buffer full
1044  05F9 E823FB        011F    CALL    diskrc
1045  05FC 5B                    POP     BX                  ;recover address of bptr
1046  05FD 751B          061A    JNZ     typeof              ;hard end of file
1047  05FF BB0000                mov     bx,0h               ;bptr = 0
1048                       type1:                           ;read character at bptr and print
1049  0602 43                    inc     bx                  ;bptr = bptr + 1
1050  0603 BE4A08                mov     si,offset ccpbuff-1
1051  0606 03F3                  add     si,bx
1052  0608 8A04                  mov     al,byte ptr [si]
1053  060A 3C1A                  cmp     al,eofile
1054  060C 7410          061E    jz      jretcom
1055  060E 53                    push    bx
1056  060F E886FA        0098    CALL    printchar
1057  0612 E8B4FB        01C9    CALL    breakk
1058  0615 5B                    pop     bx
1059  0616 7506          061E    jnz     jretcom
1060                                                         ;abort if break
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
1061
1062  0618 EBD8        05F2        jmps    type0               ;for another character
1063                                                          ;
1064                         typeof:                          ;end of file, check for errors
1065  061A FEC8                    DEC     AL
1066  061C 7503        0621        jnz     typerr
1067                         jretcom:
1068  061E E9C001      07E1        JMP     retcom
1069                         typerr:
1070  0621 E891FD      03B5        CALL    readerr
1071  0624 E845FE      046C        CALL    resetdisk
1072                         jcomer:
1073  0627 E996FD      03C0        jmp     comer
1074                         ;
1075                         ;
1076                         rename:
1077                                                          ;rename a file on a specific disk
1078  062A 2EC6060A0000            mov     comlen,0h           ;user may abort disk error
1079                                                          ;with control-C
1080  0630 E83AFC      026D        call    filfcb0
1081  0633 75F2        0627        jnz     jcomer
1082                                                          ;must be unambiguous
1083  0635 A03A09                  MOV     AL,sdisk
1084  0638 50                      PUSH    AX                  ;save for later compare
1085  0639 E810FE      044C        CALL    setdsk              ;disk selected
1086  063C E8CDFA      010C        CALL    srchcm              ;is new name already there?
1087  063F 7402        0643        JZ      L_47
1088  0641 EB55        0698        jmps    rener3
1089                         L_47:
1090                                                          ;file doesn't exist, move to second half of f
1091  0643 BE2708                  mov     si,offset comfcb
1092  0646 BF3708                  mov     di,offset comfcb+16
1093  0649 B91000                  MOV     cx,16
1094  064C E8EDFD      043C        CALL    move0
1095                                                          ;check for = or left arrow
1096  064F 8B363509                mov     si,comaddr
1097  0653 E8E3FB      0239        CALL    deblank
1098  0656 3C3D                    CMP     AL,'='
1099  0658 7404        065E        JZ      ren1                ;ok if =
1100  065A 3C5F                    CMP     AL,1a
1101  065C 7535        0693        JNZ     rener2
1102                         ren1:
1103  065E 46                      inc     si
1104  065F 89363509                mov     comaddr,si          ;past delimiter
1105                                                          ;proper delimiter found
1106  0663 E807FC      026D        CALL    filfcb0
1107  0666 752B        0693        JNZ     rener2
1108                                                          ;check for drive conflict
1109  0668 58                      POP     AX
1110  0669 8AC8                    MOV     CH,AL               ;previous drive number
1111  066B BB3A09                  mov     bx,offset sdisk
1112  066E 8A07                    MOV     AL,BYTE PTR [BX]
1113  0670 0AC0                    OR      AL,AL
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
1114
1115  0672 7406          067A          JZ      ren2
1116                                                    ;drive name was specified.  same one?
1117  0674 3AC5                        CMP     AL,CH
1118  0676 882F                        MOV     BYTE PTR [BX],CH
1119  0678 7519          0693          JNZ     rener2
1120  067A 882F               ren2:    MOV     BYTE PTR [BX],CH        ;store the name in case drives switch
1121  067C C606270800                  mov     byte ptr comfcb,0h
1122  0681 E888FA        010C          CALL    srchcm                  ;is old file there?
1123  0684 7408          068E          JZ      rener1
1124                                                    ;
1125                                                    ;everything is ok, rename the file
1126  0686 BA2708                      mov     dx,offset comfcb
1127  0689 E89EFA        012A          CALL    renam
1128                          jp_retcom:
1129  068C EB90          061E          JMPS    jretcom
1130                                                    ;
1131                          rener1:                   ; no file on disk
1132  068E E829FD        03BA          CALL    nofile
1133  0691 EBF9          068C          JMPS    jpretcom
1134                          rener2:                   ; ambigous reference/name conflict
1135  0693 E8D6FD        046C          CALL    resetdisk
1136  0696 EB38          06D0          jmps    ujmperr
1137                          rener3:                   ; file already exists
1138  0698 B96F09                      mov     cx,offset renmsg
1139  069B E811FA        00AF          CALL    print
1140  069E EBEC          068C          JMPS    jpretcom
1141                                ;
1142                                ;
1143                          user:
1144                                                    ;set user number
1145  06A0 E847FD        03EA          call    getnumber               ; leaves the value in the accumulator
1146  06A3 3C21                        cmp     al,crr                  ;is this a request for user no.
1147  06A5 751A          06C1          jnz     outuser
1148  06A7 E8FCF9        00A6          call    crlf
1149  06AA E881FA        012E          call    getuser                 ;find current user
1150  06AD 0C30                        or      al,30h                  ;make this ascii for printchar
1151  06AF 3C39                        cmp     al,39h                  ;check for numbers 10-16
1152  06B1 7609          06BC          jna     user2
1153                          user1:
1154  06B3 50                          push    ax
1155  06B4 8031                        mov     al,asone                ;print two digits
1156  06B6 E8DFF9        0098          call    printchar
1157  06B9 58                          pop     ax
1158  06BA 2C0A                        sub     al,0ah                  ;correction to second digit
1159  06BC E8D9F9        0098 user2:   call    printchar
1160  06BF EB49          070A          jmps    jpendcom
1161                                ;
1162                          outuser:
1163  06C1 3C10                        cmp     al,16                   ;legal user number??
1164  06C3 7202          06C7          jb      outuser1
1165  06C5 EB09          06D0          jmps    ujmperr
1166                          outuser1:                 ; must be between 0 and 15
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER #

```
1167
1168  06C7 8AD0                 mov     dl,al                   ;save for setuser call
1169  06C9 803E280820           cmp     byte ptr comfcb+1,' '
1170  06CE 7503         06D3    jnz     outuser2
1171                        ujmperr:
1172  06D0 E9EDFC       03C0    jmp     comer
1173                        outuser2:
1174  06D3 A2CE08               mov     cuser,al                ;save new user number for abort
1175  06D6 E857FA       0130    CALL    setuser                 ;new user number set
1176  06D9 E90801       07E4    JMP     endcom
1177                        ;
1178                        ;
1179                        ;
1180                        ;
1181                        ;
1182                        ;
1183                        loaderr:                            ;cannot load the program
1184  06DC B97B09               mov     cx,offset loadmsg
1185  06DF E8CDF9       00AF    CALL    print
1186  06E2 EBA3         068C    JMPs    jpretcom
1187                        ;
1188                        ;
1189                        ;  page 0 equates
1190    0003                usercode        equ     word ptr 3
1191    0005                useripO equ     byte ptr 5
1192                        ;
1193                        ;
1194                        nxtbase:
1195
1196                        ;
1197                        ;
1198    000F                esbase  equ     0fh             ;base of EXTRA seg in the base page
1199                        ;
1200                        userfunc:
1201  06E4 E827FB       020E    CALL    serize                  ;check serialization
1202                                                            ;load user function and set up for execution
1203  06E7 803E280820           cmp     byte ptr comfcb+1,' '
1204  06EC 751F         070D    JNZ     user0
1205                                                    ;no file name, but may be disk switch
1206  06EE A03A09               mov     al,sdisk
1207  06F1 0AC0                 or      al,al
1208  06F3 7415         070A    jz      jpendcom
1209                        chng_driv:                          ;no disk name if 0
1210  06F5 FEC8                 dec     al
1211  06F7 A23709               mov     cdisk,al
1212  06FA A33109               mov     cxpar,ax                ;call BIOS to check valid drive
1213  06FD E8CAF9       00CA    call    biosfunc
1214  0700 0BDB                 or      bx,bx                   ;BX is zero for invalid drive
1215  0702 74CC         06D0    jz      ujmperr
1216  0704 A03909               mov     al,cdisk
1217  0707 E8D0F9       00DA    call    select
1218                        jpendcom:
1219  070A E9D700       07E4    jmp     endcom
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
1220
1221                            user0:                          ;file name is present
1222  070D 803E300820                   cmp     comfcb+9,' '    ;is filetype present??
1223  0712 7402           0716         jz      userf1
1224  0714 EBBA           06D0         jmps    ujmperr
1225                            userf1:                         ;type is blank
1226  0716 E833FD         044C         call    setdsk
1227  0719 BF3008                      mov     di,offset comfcb+9
1228  071C BE0008                      mov     si,offset cmdtype
1229                                             ;file type is set to .cmd
1230  071F B90300                      mov     cx,3
1231  0722 F3A4                 rep    movs    al,al
1232  0724 BA2708                      mov     dx,offset comfcb
1233  0727 E8C3F9         00ED         call    openc
1234  072A 7506           0732         jnz     loaduser
1235  072C E83DFD         046C         call    resetdisk
1236                            jpcomer:
1237  072F E98CFC         03C0         jmp     comer
1238                            loaduser:
1239  0732 E88EF9         00C3         call    ldfile
1240  0735 40                          inc     ax
1241  0736 74A4           06DC         jz      loaderr
1242                                                            ;file opened properly, read it into memory
1243                            gouser:
1244  0738 891E0308                    mov     pag0,bx         ;address of pag0
1245  073C 8EC3                        mov     es,bx
1246  073E E82BFD         046C         CALL    resetdisk       ;back to original disk
1247  0741 8CD8                        mov     ax,ds
1248  0743 8EC0                        mov     es,ax           ;restore the es register
1249  0745 E825FB         026D         CALL    filfcb0
1250  0748 BB3A09                      mov     bx,offset sdisk
1251  074B 53                          PUSH    BX
1252  074C 8A07                        MOV     AL,BYTE PTR [BX]
1253  074E A22708                      MOV     comfcb,AL       ;drive number set
1254  0751 B81000                      mov     ax,16
1255  0754 E819FB         0270         CALL    filfcb          ;move entire fcb to memory
1256  0757 5B                          POP     BX
1257  0758 8A07                        MOV     AL,BYTE PTR [BX]
1258  075A A23708                      MOV     comfcb+16,AL
1259  075D C606470800                  mov     byte ptr comrec,0h      ;record number set to zero
1260  0762 BF5C00                      mov     di,userfcb
1261  0765 8E060308                    mov     es,pag0         ;move fcb info to user page0
1262  0769 BE2708                      mov     si,offset comfcb
1263  076C B92100                      mov     cx,33
1264  076F E8CAFC         043C         CALL    move0
1265                                                            ;move command line to buff
1266  0772 BE0B00                      mov     si,offset combuf
1267  0775 E8C1FA         0239         call    deblank
1268  0778 0AC0                        or      al,al
1269  077A 740E           078A         jz      bmove1
1270  077C AC                 bmove0:  lods    al
1271  077D 0AC0                        or      al,al
1272  077F 7409           078A         jz      bmove1
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
1273
1274  0781 3C20              CMP     AL,' '
1275  0783 7402        0787  jz      bmovea
1276  0785 EBF5        077C  jmps    bmove0          ;for another scan
1277                                                 ;first blank position found
1278                   bmovea:
1279  0787 E8AFFA      0239  call    deblank
1280  078A B500        bmove1: MOV   CH,0
1281  078C BF8100            mov     di,userbuff+1
1282                                                 ;ready for the move
1283                   ;
1284  078F 4E                dec     si
1285                   ;
1286                   bmove2:
1287  0790 AC                lods    al
1288  0791 AA                stos    al
1289  0792 0AC0              OR      AL,AL
1290  0794 7404        079A  JZ      bmove3
1291                                                 ;more to move
1292  0796 FEC5              INC     CH
1293  0798 EBF6        0790  jmps    bmove2
1294                   bmove3:                       ;b has character count
1295  079A 8AC5              MOV     AL,CH
1296  079C BF8000            mov     di,userbuff
1297  079F AA                stos    al             ;store buffer count
1298  07A0 BF0000            mov     di,0
1299  07A3 268B4503          mov     ax,es:usercode[di]
1300  07A7 2EA3FB07          mov     cspage,ax      ;set up pointer for far call
1301  07AB 268A4505          mov     al,es:userip0[di]
1302  07AF 2EA2FA07          mov     byte ptr ip0+1,al
1303  07B3 E8F0F8      00A6  CALL    crlf
1304                                                 ;now go to the loaded program
1305  07B6 E82AFA      01E3  CALL    setbufu        ;default user dma
1306                                                 ;low memory disks contains user code
1307  07B9 C606CF0800        mov     bptr,0h        ;buffer pointer is zero
1308  07BE 2EC6060A0000      mov     comlen,00h     ;zero command length
1309  07C4 8E060308          mov     es,pag0
1310  07C8 8E1E0308          mov     ds,pag0        ;set user's ds register
1311  07CC 8B0F00            mov     bx,esbase
1312  07CF 833F00            cmp     word ptr[bx],0 ;did user have extra segment
1313  07D2 7402        07D6  jz      calluser
1314  07D4 8E07              mov     es,[bx]
1315  07D6 2EFF1EF907  calluser:     callf   dword ptr ip0        ;gone to the loaded program
1316  07DB BC3009            mov     sp,offset stack_base
1317  07DE E931F8      0092  jmp     funct0
1318                                                 ;
1319                   ;
1320                                                 ;
1321                   ;
1322                   ;
1323                   retcom:                       ;reset disk before end of command check
1324  07E1 E888FC      046C  CALL    resetdisk
1325                   ;
```

CCP/M-86 © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
1326
1327                            endcom:                         ;end of intrinsic command
1328  07E4 E886FA       026D        CALL    filfcb0             ;to check for garbage at end of line
1329  07E7 A02808                   MOV     AL,comfcb+1
1330  07EA 2C20                     SUB     AL,' '
1331  07EC BB3A09                   mov     bx,offset sdisk
1332  07EF 0A07                     OR      AL,BYTE PTR [BX]
1333                                                            ;0 in accumulator if no disk selected, and bl
1334  07F1 7503         07F6        jnz     ejpcomer
1335  07F3 E968FB       035E        jmp     ccp
1336                            ejpcomer:
1337  07F6 E9C7FB       03C0        jmp     comer
1338                            ;
1339                            ;
1340                            ;
1341                            ;
1342  07F9 0000                 ip0     dw      0       ;will hold initial program start address
1343  07FB 0000                 cspage  dw      0       ;base of code area
1344                            ;
1345                            ;
1346                                    dseg
1347                                    org     ccpcode
1348                            ;
1349                            ;       data areas
1350                            ;
1351                            ;
1352                            ;
1353                            ;
1354  0800 434D44               cmdtype db      'CMD'           ;for absolute image files
1355  0803 0000                 pag0    dw      0       ;used to store base of user data group
1356                            ;
1357                            ;
1358                            ;
1359                            ;       'submit' file control block
1360  0805 00                   submit  db      0                       ;00 if no submit file, ff if submitting
1361  0806 002424242020         subfcb  db      0,'$$$     '            ;file name is $$$
1362       202020
1363  080F 5355420000                   db      'SUB',0,0               ;file type is sub
1364  0814 00                   submod  db      0                       ;module number
1365  0815                      subrc   rs      1               ;record count filed
1366  0816                              rs      16                      ;disk map
1367  0826                      subcr   rs      1                       ;current record to read
1368                            ;
1369                            ;       command file control block
1370  0827                      comfcb  rs      32
1371  0847                      comrec  rs      1               ;currect record to read/write
1372  0848                              rs      3
1373  084B                      ccpbuff rs      130     ;ccp buffer space for type and dir
1374  08CD                      dcnt    rs      1               ;disk directory count (used for error codes)
1375  08CE                      cuser   rs      1               ;current user
1376                                                                    ;none=0, a=1, b=2 ...
1377  08CF                      bptr    rs      1               ;buffer pointer
1378  08D0                              rs      96      ;working stack
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
1379
1380  0930                    stack_BASE      rs      0
1381                          ;
1382                          ;
1383  0930 09                 biosparm        db      9
1384  0931 0000               cxpar           dw      0
1385  0933 0100               dxpar           dw      1
1386                          ;
1387                          ;
1388  0935 0B00               comaddr dw      offset(combuf)
1389  0937 0000               staddr  dw      0               ;starting address of fill fcb request
1390  0939                    cdisk   rs      1
1391  093A 00                 sdisk   db      0
1392  093B 203D5F2E3A3B       delims  db      ' ','=',5fh,',','.',':',';','<','>',',','
1393       3C3E2C
1394  0944 44495253           intvec  db      'DIRS'
1395  0948 55534552                   db      'USER'
1396  094C 52454E20                   db      'REN '
1397  0950 54595045                   db      'TYPE'
1398  0954 45524120                   db      'ERA '
1399  0958 44495220                   db      'DIR '
1400    0006                  intlen  EQU     6               ;intrinsic function length
1401  095C 4E4F2046494C       nofmsg  db      'NO FILE',0
1402       4500
1403  0964 414C4C202859       ermsg   db      'ALL (Y/N)?',0
1404       2F4E293F00
1405  096F 46494C452045       renmsg  db      'FILE EXISTS',0
1406       584953545300
1407  097B 4D454D4F5259       loadmsg db      'MEMORY NOT AVAILABLE',0
1408       204E4F542041
1409       5641494C4142
1410       4C4500
1411  0990 524541442045       rdmsg   db      'READ ERROR',0
1412       52524F5200
1413  099B 0D0A43414E4E       closem          db      cr,lf,'CANNOT CLOSE',0
1414       4F5420434C4F
1415       534500
1416  09AA 0D0A4E4F4E2D       dirfiles        db      cr,lf,'NON-SYSTEM FILE(S) EXIST',0
1417       535953544D
1418       2046494C4528
1419       532920455849
1420       535400
1421  09C5 0D0A53595354       sysfiles        db      cr,lf,'SYSTEM FILE(S) EXIST',0
1422       4D2046494C
1423       452853292045
1424       5849535400
1425                          ;
1426  09DC 000000000000       serial  db      0,0,0,0,0,0
1427                          ;
1428                          ;pc changes to the proper intrinsic or user function
1429  09E2 9A04               jmptab  dw      OFFSET(direct)          ;directory search
1430  09E4 8A05                       dw      OFFSET(erase)           ;file erase
1431  09E6 D205                       dw      OFFSET(typec)           ;type file
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
1432
1433  09E8 2A06                    dw      OFFSET(rename)          ;file rename
1434  09EA A006                    dw      OFFSET(user)            ;user number
1435  09EC 8304                    dw      OFFSET(directs)
1436  09EE E406                    dw      OFFSET(userfunc)        ;user-defined function
1437  09F0 00              anyfils db      0
1438  09F1 00              typedir db      0                       ;type of directory display
1439  09F2                 dmaoff  rw      1       ;offset of 20 bit DMA address
1440  09F4                 dmaseg  rw      1       ;segment of 20-bit DMA address
1441  09F6 0000            mcbbase dw      0       ;base of memory image for save
1442  09F8 0000            mcblen  dw      0       ;length of memory image
1443                       ;
1444                                org     ccpcode+250h
1445  0A50 446563656D62            db      'December 28 81'
1446       657220323320
1447       3831
1448                                end     ;END
1449
1450
1451  END OF ASSEMBLY.  NUMBER OF ERRORS:  0.  USE FACTOR: 20%
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950

SER. #_____

```
ADDHCF          0440 L  790#  898   948   959
ANYFILS         09F0 V  842   854   865   906   912  1437#
ASONE           0031 N   20#  1155
BADSERIAL       021E L  403   406#
BDOS            008B L  111#  119   124   164   171   175   178   183   186   230
                        262   324   345   351   376
BDOSL           0B00 L   14#  379
BIOSF           0032 N   87#  169
BIOSFUNC        00CA L  163#  1213
BIOSPARM        0930 V  170   1383#
BLANK           009E L  126#  940   944   974
BLANKE          0020 N  433#  444   596
BMOVE0          077C L  1270#  1276
BMOVE1          078A L  1269  1272  1280#
BMOVE2          0790 L  1286#  1293
BMOVE3          079A L  1290  1294#
BMOVEA          0787 L  1275  1278#
BOSINK          00E0 L  185#  193   202   212   218
BPTR            08CF V  1307  1377#
BREAKF          000B N   70#  344
BREAKK          01C9 L  312   342#  979   1057
CALLNOFILE      05CD L  1019#  1034
CALLUSER        07D6 L  1313  1315#
CCP             035E L  315   658#  675   678   731   1000  1005  1335
CCP0            0378 L  656   670#
CCP2            039F L  683   685#
CCPBUFF         034B V  298   606   791   1050  1373#
CCPCLEAR        0327 L   45   624#
CCPCODE         0800 N   15#  1347  1444
CCPLOC          0000 L   13#   14
CCPRET          030B L   46   614#
CCPRET1         031F L  617   619#
CCPSTART        032D L   44   628#
CCPSTART1       0353 L  622   651#
CDISK           0939 V  281   308   393   512   648   673   807   826   831   1211
                        1216  1390#
CHNGDRIV        06F5 L  1209#
CLOSE           00F7 L  200#  305
CLOSEERR        00FF L  203   205#
CLOSEF          0010 N   75#  201
CLOSEM          099B V  206   1413#
CMDTYPE         0800 V  1228  1354#
COMADDR         0935 V  339   504   558   615   676   1010  1096  1104  1388#
COMBUF          000B V   50#  310   339   615   616   674   1266  1388
COMER           03C0 L  422   680   687   710#  736   754   758   764   766   769
                        777   869   1073  1172  1237  1337
COMER0          03C7 L  715#  725
COMER1          03D9 L  718   720   726#
COMFCB          0327 V  162   197   222   241   500   583   686   739   801   872
                        1012  1015  1030  1091  1092  1121  1126  1169  1203  1222
                        1227  1232  1253  1258  1262  1329  1370#
COMLEN          000A V   48#  297   326   328   618   626   655   730   864   989
                        997   1026  1078  1308
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
COMREC          0847 V   196  1259  1371#
CONV0           0402 L   742   747#  773
CONV1           042F L   749   775#  781
CR              000D N    97#  137  1413  1416  1421
CRLF            00A6 L   136#  145   662   713   729   930  1036  1148  1303
CRR             0021 N    96#  744  1146
CSELECT         01DB L   355#  663   672   932
CSELF           0019 N    84#  357
CSPAGE          07FB V  1300  1343#
CTLC            0003 N    19#  616
CUSER           08CE V   639   815  1174  1375#
CXPAR           0931 V  1212  1384#
DCNT            08CD V   187   892  1374#
DEBLANK         0239 L   435#  449   505   677  1097  1267  1279
DEBLANK0        0240 L   441   443#
DEBLANK1        0245 L   445   447#
DELETE          0111 L   225#  392  1016
DELF            0013 N    78#  226
DELIM           021F L   414#  467   482
DELIM0          0226 L   417   419#
DELIM1          022B L   421   423#
DELIMS          093B V   426  1392#
DELSUB          01F2 L   314   318   382#
DELSUB1         01FA L   385   387#
DIR             04B2 L   841   853   863#
DIR0            04D5 L   877#  880
DIR0A           04C8 L   868   870#
DIR1            04DD L   874   882#
DIR2            04E8 L   885   888#  982
DIR21           04ED L   888   890#
DIR25           051B L   905   911   916#
DIR3            054A L   947#  972   975
DIR4            0568 L   952   965#
DIR5            057C L   962   969   977#
DIR6            057D L   907   913   979#
DIRB            0566 L   957   964#
DIRECT          049A L   850# 1429
DIRECT2         04AF L   847   855   858#
DIRECTS         0483 L   838# 1435
DIRECTS2        0498 L   843   846#
DIRFILES        09AA V   844  1416#
DIRHD0          053D L   929   940#
DIRHD1          0545 L   939   943#
DISKR           011B L   234#  242   293
DISKRC          011F L   239# 1044
DMABASF         0033 N    88#  361
DMAF            001A N    85#  374
DMAOFF          09F2 V  1439#
DMASEG          09F4 V  1440#
DREADF          0014 N    79#  236
DWRITF          0015 N    80#
DXPAR           0933 V  1385#
EFILL           02BA L   550#
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950

SER #_____

```
EJPCOMER        07F6 L  1334  1336#
ENDCOM          07E4 L  1176  1219  1327#
ENDGETID        026C L   483   486#
ENDIR           0588 L   889   980   983#
EOFILE          001A N   100#  1053
ERARE1          05D0 L  1018  1021#
ERASE           058A L   988#  1430
ERASEFILE       05B9 L   992  1011#
ERMSG           0964 V   994  1403#
ERRM            0124 L   237   245#
ESBASE          000F N  1198#  1311
FALSE           0000 N    16#   17    18
FILER           029D L   516   520#
FILFCB          0270 L   499#  1255
FILFCB0         026D L   496#   679   734   866   990  1027  1080  1106  1249  1328
FILFEXIT        02DB L   5[illegible]   572#
FUNCT0          0092 L   116#  1317
GETID1          025F L   472   475#
GETID2          0260 L   474   477#
GETIDENT        024F L   461#   480   536   545
GETNUMBER       03EA L   733#  1145
GETUSER         012E L   255#  1149
GOUSER          0738 L  1243#
INITF           000D N    72#   177
INITIALIZE      00D6 L   176#   643
INITNAF         003A N    89#   174
INITNEW         00D2 L   173#   645
INTLEN          0006 N   581  1400#
INTR1           02E4 L   582#   590
INTR11          02F3 L   591#   597
INTR2           02F6 L   587   594#
INTRINSIC       02DE L   577#   689
INTVEC          0944 V   580  1394#
IP0             07F9 V  1302  1315  1342#
JCOMER          0627 L  1013  1028  1031  1072#  1081
JMPTAB          09E2 V   692  1429#
JPBDOS3         01EF L   358   362   375#
JPCOMER         072F L  1236#
JPENDCOM        070A L  1160  1208  1218#
JPRETCOM        068C L  1128#  1133  1140  1186
JPSELECT        0480 L   811   832#
JRETCOM         061E L  1022  1054  1059  1067#  1129
L1              00BB L   149   151#
L29             0428 L   770#
L33RET          047C L   804   809   824   829#
L38             05AB L   999  1001#
L39             05B4 L  1004  1006#
L47             0643 L  1087  1089#
L6              01D3 L   347   349#
LA              005F N    99#  1100
LF              000A N    98#   139  1413  1416  1421
LIFTF           000C N    71#
LOADERR         06DC L  1183#  1241
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
LOADMSG          097B V  1184  1407#
LOADUSER         0732 L  1234  1238#
LODFILE          00C3 L   161# 1239
LOGF             0018 N    83#
LRET4            013F L   269   272   274#
MAKEF            0016 N    81#
MAXLEN           0009 V    47#  323
MCBBASE          09F6 V  1441#
MCBLEN           09F8 V  1442#
MOVE0            043C L   300   785# 1094  1264
NOFILE           03BA L   705#  886  1020  1132
NOFMSG           095C V   707  1401#
NONSYSFILES      050D L   900   909#
NOREAD           01AA L   313   325#
NOSUB            019F L   279   288   294   306   317#
NXTBASE          06E4 L  1194#
OPEN             00E9 L   191#  198   287
OPENC            00ED L   195# 1033  1233
OPENF            000F N    74#  172
OSCOND           0115 L   229#  237
OUTUSER          06C1 L  1147  1162#
OUTUSER1         06C7 L  1164  1166#
OUTUSER2         06D3 L  1170  1173#
PADFIELD         0248 L   453#  458   469
PAGO             0803 V   367  1244  1261  1309  1310  1355#
PBUFF            0009 N    68#
PCHARF           0002 N    67#  123
PRINO            00B4 L   147#  156   311
PRINT            00AF L   143#  246   709   845   857   995  1139  1185
PRINTBC          00A0 L   129#  138   140   936   938   942   966
PRINTCHAR        0098 L   121#  132   154   665   667   722   728  1056  1156  1159
RBUFF            000A N    69#  322
RCHARF           0001 N    66#  350
RDMSG            0990 V   702  1411#
READC            0140 L   276#  669   996
READC0           01B6 L   330#  335
READC1           01C0 L   329   337#
READCA           0153 L   282   285#
READERR          03B5 L   700# 1070
REN1             065E L  1099  1102#
REN2             067A L  1115  1120#
RENAM            012A L   251# 1127
RENAME           062A L  1076# 1433
RENER1           068E L  1123  1131#
RENER2           0693 L  1101  1107  1119  1134#
RENER3           0698 L  1088  1137#
RENF             0017 N    82#  252
RENMSG           096F V  1138  1405#
RESET1           047D L   817   828   831#
RESETDISK        046C L   820# 1071  1135  1235  1246  1324
RESETDU          0463 L   621   812#
RETCOM           07E1 L   247   859  1068  1323#
ROFILE           0009 N    72#
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950

SER #_____

```
SCNQ            02CF L   564#  569
SCNQO           02D7 L   567   569#
SDISK           073A V   503   526   682   735   802   822  1083  1111  1206  1250
                        1331  1391#
SEARF           0011 N    76#  211
SEARNF          0012 N    77#  217
SELECT          00DA L   180#  284   309   390   394   649   833  1217
SELF            000E N    73#  182
SERO            021D L   402   404#
SERIAL          09DC V   398  1426#
SERIZE          020E L   396# 1201
SETBUFU         01E3 L   364# 1305
SETCCPDMA       02FF L   603#  653   658
SETCRO          0290 L   507   511#
SETDMA          01ED L   371#  607
SETDMAB         01DF L   359#  368   605
SETDSC          0295 L   510   514#
SETDSC1         02A2 L   518   525#
SETDSCX         02A7 L   513   529#
SETDSK          044C L   799#  871  1014  1032  1085  1226
SETTY1          02B7 L   542   544#
SETUSER         0130 L   259#  641   816  1175
SRCH            0104 L   210#  223
SRCHCM          010C L   220#  884  1086  1122
SRCHN           0108 L   215#  931
STACKBASE       0930 V   620   630   661  1316  1380#
STADDR          0937 V   506   714  1389#
SUBCR           0826 V   291  1367#
SUBFC3          0806 V   286   292   304   391  1361#
SUBNIT          0805 V   278   384   389   644  1360#
SUBNOD          0814 V   302  1364#
SUBRC           0815 V   289   303  1365#
SYSFILE         000A N    93#  897
SYSFILES        09C5 V   856  1421#
TESTING         0000 N    13#
TOBDOS2         0132 L   227   253   261#
TOPRINT         03BD L   703   708#
TRANSLATE       0135 L   265#  332
TRID            0264 L   481#  485
TRUE            FFFF N    17#  904   910
TYPEO           05F2 L  1038# 1062
TYPE1           0602 L  1041  1048#
TYPEC           05D2 L  1025# 1431
TYPEDIR         09F1 V   840   852   904   910  1438#
TYPEOF          061A L  1046  1064#
TYPERR          0621 L  1066  1069#
UJMPERR         06D0 L  1136  1165  1171# 1215  1224
USER            06A0 L  1143# 1434
USERO           070D L  1204  1221#
USER1           06B3 L  1153#
USER2           06BC L  1152  1159#
USERBUFF        0080 N   108#  369  1281  1296
USERCODE        0003 N  1190# 1299
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950

SER #_____

USERF	0020	N	86#	260
USERF1	0716	L	1223	1225#
USERFCB	005C	N	106#	1260
USERFUNC	06E4	L	684	1200# 1436
USERIP0	0005	N	1191#	1301

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950

SER. #_____